FACULTAD DE CIENCIAS ECONOMICAS Y EMPRESARIALES
BIBLIOTECA
UCM

W
28
(9004)

Documento de Trabajo

9 0 0 4

SHIFT ASSIGNMENT BY INTEGER PROGRAMMING TECHNIQUES

Miguel Jerez Méndez
Daniel Villalba Vila

FACULTAD DE CIENCIAS ECONOMICAS Y EMPRESARIALES
BIBLIOTECA

FACULTAD DE CIENCIAS ECONOMICAS Y EMPRESARIALES.- UNIVERSIDAD COMPLUTENSE
Campus de Somosaguas. 28023 - MADRID

Esta publicación de Documentos de Trabajo pretende ser cauce de expresión y comunicación de los resultados de los proyectos de investigación que se llevan a cabo en la Facultad de Ciencias Económicas y Empresariales de la Universidad Complutense de Madrid. No obstante, la publicación está abierta a investigadores de otras instituciones que deseen difundir sus trabajos en ella.

Los Documentos de Trabajo se distribuyen gratuitamente a las Universidades e Instituciones de Investigación que lo soliciten. Asímismo, las peticiones personales pueden ser atendidas en la medida en que se disponga de ejemplares en existencia.

Se ruega a las personas e instituciones interesadas en solicitar ejemplares que utilicen el boletín de pedido que figura seguidamente.

DOCUMENTOS DE TRABAJO

Boletín de Pedido.

Nombre de la persona o institución: ....

....

Calle: .... nº ....

Ciudad: ....Distrito Postal:.........País: ....

Solicita una suscripción permantente ....[ ]

(sólo Universidades e Instituciones de Investigación)

Solicita los Documentos de Trabajo cuyos números se relacionan a continuación: ____________________

____

Enviar a:

Facultad de Ciencias Económicas y Empresariales

Universidad Complutense de Madrid

Vicedecano

Campus de Somosaguas. 28023 MADRID. ESPAÑA.

Shift Assignment by Integer Programming Techniques

Miguel Jerez Méndez*
Daniel Villalba Vila**

ABSTRACT: In this paper we describe a Decision Support System (DSS) designed to solve shift assignment problems by means of integer programming. It is based on a mathematical model which extends previous formulations in several ways. This DSS has been applied successfully to real problems in a corporate environment.

---

* Dept. of Quantitative Economics, Complutense University of Madrid, School of Economics.

** Dept. of Business Administration, Autónoma University of Madrid, School of Economics.

Both authors can be reached for non academical purposes at Decision Support Systems S.A., C/ Marbella 35-1, 28034 MADRID (SPAIN).

We are very grateful to 'El Corte Inglés' personnel studies staff. Without their knowledgeable help and professionality this article couldn't have been written.

1. Introduction.

In this paper we describe an integer programming model designed to solve shift assignment problems. It is the nucleus of an interactive DSS called **PETS**[1]. **PETS** has been applied at 'El Corte Inglés', a big commercial corporation with over 30,000 salespeople, yielding substantial savings in personnel costs and improving the quality of service.

The main characteristics of the optimization problem are:

- **Objective:** optimize a linear combination of service quality and cost[2].
- **Decision variables:** number of employees to be assigned to each shift.
- **Constraints:** a) cover the workload and b) satisfy a set of arbitrary constraints (maximum number of employees, agreements between the firm and the workers, etc.)

This kind of problems does not have a long tradition in the Operations Research literature. However, one example is Jaquet-Lagrèze and Meziani (1988).

In Section 2 we discuss several formulations of the single department shift assignment problem. Section 3 describes an extension which allows for a shared personnel pool. Sample cases are provided for both models to illustrate some of its applications. Several practical issues, concerning the implementation of the system, are addressed in Section 4. The conclusions are summarized in Section 5.

---

[1] **PETS** stands for **PE**rsonnel **T**ime **S**cheduling.

[2] Another valid approach would be to optimize sequentially both goals (multicriterion approach). We will use the multiobjective approach for reasons of simplicity.

2. Mathematical models for the shift assignment problem.

Let us consider a firm which conforms with the following assumptions:

1) All the personnel subject to the shift assignment process can be considered as homogeneous units of manpower.

2) The target of the decisionmaker is twofold:
   2.a Minimize personnel costs.
   2.b Adjust the shift plan to the workload of each department, so that in every period there is a number of active employees greater or equal to the requirements.

   We will also assume that the decisionmaker is able to quantify the trade-offs among both goals.

3) The set of possible shifts is finite and explicitly defined.

Furthermore, we will assume that the organization can be considered either a single unit or divided in several very specialized departments, so that personnel swaps are not part of the decision problem. A method to relax this premise is discussed in Section 3.

Under these hypotheses, we will describe three shift assignment models.

2.1. The hard constraints model.

By assumption 1, the decisionmaker is interested in minimizing the total cost of his shift schedule. At the same time, he wants to maintain a reasonable level of quality in the attention to customers. This quality requirement could be modelled by a set of constraints which requires the coverage of the expected workload at every time period in a given day. In this situation, the following model could be used:

Model 1:

$$\text{Min} \quad \sum_{h \in H} c_h \, x_h$$

subject to:

$$\sum_{h \in H} a_{ht} \, x_h \geq \underline{X}_t \qquad (t \in T)$$

$$x_h \in \{0, 1, 2, \ldots\} \qquad (h \in H)$$

where:

- H : Set of available shifts.
- T : Set of time periods.
- c_h : Cost per employee assigned to shift **h**.
- x_h : Number of employees assigned to shift **h**.
- a_{ht} : Boolean parameter defined such that:
  a_{ht} = 1 if shift **h** is active at time **t**.
  = 0 otherwise.
- $\underline{X}_t$: Personnel requirement at time **t**.

2.2. The soft constraints model.

Obviously, in real life situations there are other constraints in effect: maximum personnel availability, agreements between the firm and the workers about the shift mix, etc. Most of these constraints may be represented by linear equations so they do not add any severe difficulties by themselves. However, they may require the violation of some of the workload-coverage constraints.

Thus, the shift assignment problem would be better represented by a 'soft constrained'[3] model like:

[3] See Lee (1972).

Model 2:

$$\text{Min} \quad \sum_{h \in H} c_h x_h + \sum_{t \in T} M_t s_t$$

subject to:

$$\sum_{h \in H} a_{ht} x_h + s_t \geq X_t \qquad (t \in T)$$

(... plus other constraints)

$$x_h \in \{0, 1, 2, \dots\} \qquad (h \in H)$$

$$s_t \in \{0, 1, 2, \dots\} \qquad (t \in T)$$

where:

s_t : Underachievement occuring at time **t**.

M_t : Penalty for the underachievement s_t.

Solutions of this problem yield a Pareto efficient[4] combination of cost and quality, according to the relative weight of both factors in the objective function.

A similar problem is discussed by Jaquet-Lagrèze and Meziani (1988). This model has been used to implement a manpower scheduling DSS at Air France: the **CHEOPS** system. **CHEOPS** is implemented in a Mainframe/Microcomputer system, and uses indirect techniques to provide integer solutions which may be suboptimal.

2.3. The extended soft constraints model.

The soft constrained model is accurate enough for many practical situations, but may yield unacceptable solutions. For example, a given solution may cover the load curve of the unit at all time intervals except one, in which the attention level drops to zero. This may occur because model 2 penalizes all the underachievements linearly.

Common solutions to this flaw are a) imposing an upper bound to all the underachievements or b) charging an extra cost

---

[4] That is, further cost savings cannot be achieved without reducing quality and, conversely, an improvement in quality cannot be achieved without raising the costs.

to the highest underachievement. A more natural and satisfactory approach is to impose an increasing cost over successive underachievements. Figure 1 shows a piecewise linear penalty criterion which could be used to this effect:

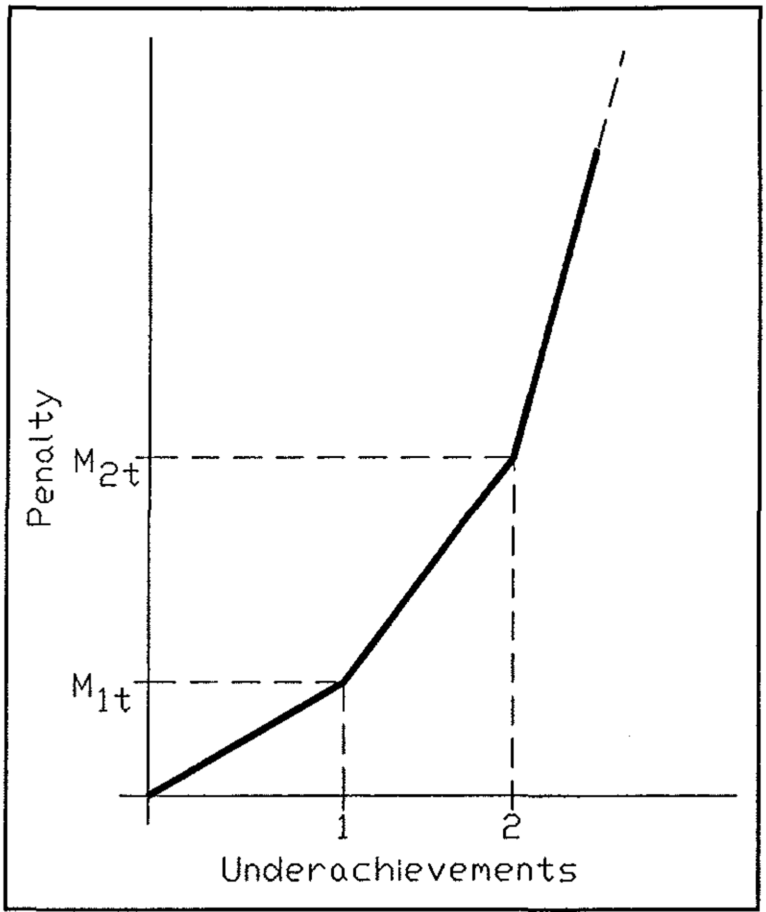

Fig. 1

In our approach, this increasing penalty is implemented by means of an n-step cost related to an n-part slack variable. These costs quantify the trade-off between cost and meeting the workload. The mathematical formulation of the model is:

Model 3:

$$\text{Min} \quad \sum_{h \in H} c_h x_h + \sum_{i=1}^{n} \sum_{t \in T} M_{it} s_{it}$$

subject to:

$$\sum_{h \in H} a_{ht} x_h + \sum_{i=1}^{n} s_{it} \geq X_t \qquad (t \in T)$$

(... plus other constraints)

$$x_h , s_{nt} \in \{0, 1, \dots \}$$

$$s_{1t}, s_{2t}, \dots, s_{n-1,t} \in \{0,1\} \qquad (h \in H) \quad (t \in T)$$

where:

s_{it} : Underachievement number i occuring at time t.

M_{it} : Penalty of the underachievement s_{it}. These weights should be fixed so that: $M_{1t} \leq M_{2t} \leq M_{3t}$.

Thus, our formulation extends the basic soft constrained model to include a nonlinear penalty.

2.4. Single department planning: the check-in counter case.

Let us consider the case of an airline which wants to optimize the working schedule of the check-in counter attendants. The following table, shows the expected workload (measured in active employees/hour) for an average day:

From:	To:		From:	To:	
00.00	01.00	2	12.00	13.00	10
01.00	02.00	2	13.00	14.00	10
02.00	03.00	2	14.00	15.00	8
03.00	04.00	2	15.00	16.00	8
04.00	05.00	2	16.00	17.00	10
05.00	06.00	3	17.00	18.00	11
06.00	07.00	3	18.00	19.00	11
07.00	08.00	4	19.00	20.00	11
08.00	09.00	4	20.00	21.00	8
09.00	10.00	5	21.00	22.00	6
10.00	11.00	6	22.00	23.00	5
11.00	12.00	8	23.00	24.00	4

Table 2.4.1: Expected workload.

We will further assume that:

- Maximum personnel availability is 17 employees.
- The agreements between the company and the workers set the following table of selectable shifts:

From:	To:	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17
00.00	01.00	1																
01.00	02.00	1	1															
02.00	03.00	1	1	1														
03.00	04.00	1	1	1	1													
04.00	05.00	1	1	1	1	1												
05.00	06.00	1	1	1	1	1	1											
06.00	07.00	1	1	1	1	1	1	1										
07.00	08.00	1	1	1	1	1	1	1	1									
08.00	09.00		1	1	1	1	1	1	1	1								
09.00	10.00			1	1	1	1	1	1	1	1							
10.00	11.00				1	1	1	1	1	1	1	1						
11.00	12.00					1	1	1	1	1	1	1	1					
12.00	13.00						1	1	1	1	1	1	1	1				
13.00	14.00							1	1	1	1	1	1	1	1			
14.00	15.00								1	1	1	1	1	1	1	1		
15.00	16.00									1	1	1	1	1	1	1	1	
16.00	17.00										1	1	1	1	1	1	1	1
17.00	18.00											1	1	1	1	1	1	1
18.00	19.00												1	1	1	1	1	1
19.00	20.00													1	1	1	1	1
20.00	21.00														1	1	1	1
21.00	22.00															1	1	1
22.00	23.00																1	1
23.00	24.00																	1

From:	To:	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18
00.00	01.00	1	1	1															
01.00	02.00	1	1	1	1	1	1												
02.00	03.00	1	1	1	1	1	1	1	1	1									
03.00	04.00	1		1	1	1	1	1	1	1	1	1	1						
04.00	05.00		1	1	1		1	1	1	1	1	1	1	1	1	1			
05.00	06.00	1	1			1	1	1		1	1	1	1	1	1	1	1	1	1
06.00	07.00	1	1	1	1	1			1	1	1		1	1	1	1	1	1	1
07.00	08.00	1	1	1	1	1	1	1	1			1	1	1		1	1	1	1
08.00	09.00	1	1	1	1	1	1	1	1	1	1	1			1	1	1		1
09.00	10.00				1	1	1	1	1	1	1	1	1	1	1			1	1
10.00	11.00							1	1	1	1	1	1	1	1	1	1	1	
11.00	12.00										1	1	1	1	1	1	1	1	1
12.00	13.00													1	1	1	1	1	1
13.00	14.00																1	1	1
14.00	15.00																		
15.00	16.00																		
16.00	17.00																		
17.00	18.00																		
18.00	19.00																		
19.00	20.00																		
20.00	21.00																		
21.00	22.00																		
22.00	23.00																		
23.00	24.00																		

From:	To:	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33	C34	C35	C36	C37	C38	C39	C40	C41	C42	C43	C44	C45	C46	C47	C48
00.00	01.00																														
01.00	02.00																														
02.00	03.00																														
03.00	04.00																														
04.00	05.00																														
05.00	06.00	1	1	1																											
06.00	07.00	1	1	1	1	1	1																								
07.00	08.00	1	1	1	1	1	1	1	1	1																					
08.00	09.00	1		1	1	1	1	1	1	1	1	1	1																		
09.00	10.00		1	1	1		1	1	1	1	1	1	1	1	1	1															
10.00	11.00	1	1			1	1	1		1	1	1	1	1	1	1	1	1	1												
11.00	12.00	1	1	1	1	1			1	1	1		1	1	1	1	1	1	1	1	1	1									
12.00	13.00	1	1	1	1	1	1	1	1			1	1	1		1	1	1	1	1	1	1	1	1	1						
13.00	14.00	1	1	1	1	1	1	1	1	1	1	1			1	1	1		1	1	1	1	1	1	1	1	1	1			
14.00	15.00				1	1	1	1	1	1	1	1	1	1	1			1	1	1		1	1	1	1	1	1	1	1	1	1
15.00	16.00							1	1	1	1	1	1	1	1	1	1	1			1	1	1		1	1	1	1	1	1	1
16.00	17.00										1	1	1	1	1	1	1	1	1	1	1			1	1	1		1	1	1	1
17.00	18.00													1	1	1	1	1	1	1	1	1	1	1			1	1	1		1
18.00	19.00																1	1	1	1	1	1	1	1	1	1	1			1	1
19.00	20.00																			1	1	1	1	1	1	1	1	1	1	1	
20.00	21.00																						1	1	1	1	1	1	1	1	1
21.00	22.00																									1	1	1	1	1	1
22.00	23.00																												1	1	1
23.00	24.00																														

Table 2.4.2: Table of shifts.

This table is structured as follows. All the shifts are coded by means of an unique three letter code. These codes (P1, P2, ..., C48) are arranged in the upper part of the table, each one corresponding to a single column. Each column contains the a_{ht} coefficients[5] defining a shift. Thus, '1' represents an active period for the shift while a blank represents an inactive period.

In these conditions, the optimal schedule is:

From:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
To:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P9	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
P15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0
P16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	0
P17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2
C3	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C11	0	0	0	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
C17	0	0	0	0	0	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
C18	0	0	0	0	0	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0
C30	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0
C33	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	0	0	0	0	0
C34	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	2	2	2	0	0	0	0
C37	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	2	2	2	0	0	0
C39	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	0	0	0
Achieved:	1	1	1	2	2	3	3	4	4	5	5	8	10	10	8	8	10	11	11	10	8	5	4	2
Desired:	2	2	2	2	2	3	3	4	4	5	6	8	10	10	8	8	10	11	11	11	8	5	4	2
Underach.	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0

Table 2.4.3: Optimal solution.

As can be seen, the solution fails to cover the load only in five hours. Also, there are no overachievements. However, this may be not an acceptable plan because from 0h to 3h the attention level is 50% of the requirement. This can be solved by increasing the underachievement penalty M_{it} at these hours. A revised solution is shown in Table 2.4.4:

[5] See Section 2.

From:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
To:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P8	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	0	0	0	0	0	0	0	0	0
P13	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
P16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	0
P17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2
C1	1	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C14	0	0	0	0	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
C30	0	0	0	0	0	0	0	0	0	2	2	2	2	2	0	2	2	2	0	0	0	0	0	0
C34	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	2	2	2	0	0	0	0
C37	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	2	0	2	2	2	2	0	0	0
C40	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1	1	1	0	0
Achieved:	2	2	2	2	2	3	3	4	4	5	5	7	10	10	8	8	10	11	10	10	7	5	4	2
Desired:	2	2	2	2	2	3	3	4	4	5	6	8	10	10	8	8	10	11	11	11	8	5	4	2
Underach.	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0

Table 2.4.4: Revised solution.

As can be seen, the cost of the revised solution remains the same (136 man-hours). Thus, underachievements have been allocated in less sensitive periods at no cost by means of an interactive planning cycle.

3. Multidepartment planning.

3.1. The multidepartment soft constraints model.

Optimization models are always abstracted from a larger problem with respect to which a given solution may be suboptimal. In the shift assignment context, a suboptimization may occur if each department is considered as a single unit isolated from the rest of the firm.

In this section we will show how to include personnel swaps between departments in the decision problem. The interest in this extension lies in the fact that several units from a firm may exhibit a complementary seasonality in their activity, so that their peak loads are distributed asynchronously. Thus, if exclusive specialization is not an issue, there may be an economic profit in pooling part of their manpower resources. In this situation, our model can be used to: a) measure the cost of not constituting such a pool and b) computing an efficient joint plan. The mathematical description of the multidepartment problem is:

$$\text{Min} \quad \sum_{h \in H} c_h \, x_h \;+\; \sum_{i=1}^{n} \sum_{d \in D} \sum_{t \in T} M_{idt} \, s_{idt}$$

subject to:

$$\sum_{h \in H} a_{hdt} \, x_h \;+\; \sum_{i=1}^{n} s_{idt} \;\geq\; \underline{X}_{dt} \qquad (d \in D) \quad (t \in T)$$

(... plus other constraints)

$$x_h \,,\; s_{ndt} \in \{0, 1, \dots\}$$

$$s_{1dt}, s_{2dt}, \dots, s_{n-1,d,t} \in \{0,1\} \qquad (h \in H) \quad (d \in D) \quad (t \in T)$$

where:

- D : Set of departments
- s_{idt} : Underachievement number **i** occuring at time **t** in department **d**.

M_{idt} : Penalty of the underachievement s_{idt}.

a_{hdt} : Boolean parameter defined such that:

a_{hdt} = 1 if shift **h** is active at time **t** in department **d**.

= 0 otherwise.

$\underline{X}_{dt}$: Personnel requirement in department **d** at time **t**.

3.2. The department store case.

Let us consider the case of a department store. The personnel requirements of two departments, measured in active salespersons/hour, are:

Hour:	Dept. #1	Dept. #2
9-10	5	8
10-11	5	8
11-12	6	8
12-13	6	7
13-14	8	6
14-15	8	5
15-16	9	4
16-17	9	4
Total	56	50

Table 3.2.1: Personnel requirements.

As can be seen, both departments exhibit complementary seasonality: peak load for the first is from 13h to 17h, while the maximum requirements of the second occur in the first four hours.

In order to simplify, let us assume that all the shifts consist in working eight hours in a row. Thus, to achieve a solution with zero underachievements, there should be 9 and 8 workers assigned to departments #1 and #2. That adds up to a total workforce of 17 salespersons. Tables 3.2.2-3 describe this plan:

From:	9	10	11	12	13	14	15	16	
To:	10	11	12	13	14	15	16	17	TOTAL
	9	9	9	9	9	9	9	9	72
Achieved:	9	9	9	9	9	9	9	9	72
Desired:	5	5	6	6	8	8	9	9	56
Overach.	4	4	3	3	1	1	0	0	16
Underach.	0	0	0	0	0	0	0	0	0

Table 3.2.2: Shift plan, department #1.

From:	9	10	11	12	13	14	15	16	
To:	10	11	12	13	14	15	16	17	TOTAL
	8	8	8	8	8	8	8	8	64
Achieved:	8	8	8	8	8	8	8	8	64
Desired:	8	8	8	7	6	5	4	4	50
Overach.	0	0	0	1	2	3	4	4	14
Underach.	0	0	0	0	0	0	0	0	0

Table 3.2.3: Shift plan, department #2.

The pooling may be modelled by means of a set of shifts which allow for an optimal allocation of the salespersons' time between the departments. These shifts are defined in the 'S' columns of the following table:

	Time	C1	C2	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
SHOP #1	9-10	1		1	1	1	1	1	1	1			
	10-11	1		1	1	1	1		1	1			
	11-12	1		1	1	1	1			1			1
	12-13	1		1	1	1	1					1	1
	13-14	1			1	1	1				1	1	1
	14-15	1				1	1				1	1	1
	15-16	1					1				1	1	1
	16-17	1									1	1	1
SHOP #2	9-10		1								1	1	1
	10-11		1					1			1	1	1
	11-12		1					1	1		1	1	
	12-13		1					1	1	1	1		
	13-14		1	1				1	1	1			
	14-15		1	1	1			1	1	1			
	15-16		1	1	1	1		1	1	1			
	16-17		1	1	1	1	1	1	1	1			

Table 3.2.4

Under these assumptions, the optimal solution is:

From:	9	10	11	12	13	14	15	16	
To:	10	11	12	13	14	15	16	17	TOTAL
C1	5	5	5	5	5	5	5	5	40
S8	0	0	0	0	2	2	2	2	8
S10	0	0	2	2	2	2	2	2	12
Achieved:	5	5	7	7	9	9	9	9	60
Desired:	5	5	6	6	8	8	9	9	56
Overach.	0	0	1	1	1	1	0	0	4
Underach.	0	0	0	0	0	0	0	0	0

Table 3.2.5: Shift plan, department #1.

From:	9	10	11	12	13	14	15	16	
To:	10	11	12	13	14	15	16	17	TOTAL
C2	6	6	6	6	6	6	6	6	48
S8	2	2	2	2	0	0	0	0	8
S10	2	2	0	0	0	0	0	0	4
Achieved:	10	10	8	8	6	6	6	6	60
Desired:	8	8	8	7	6	5	4	4	50
Overach.	2	2	0	1	0	1	2	2	10
Underach.	0	0	0	0	0	0	0	0	0

Table 3.2.6: Shift plan, department #2.

So the total workforce drops to 15 workers, saving a) 16 man-hours/day in variable costs plus b) the fixed cost of hiring two salespersons. These savings do not decrease the quality of the service, because underachievements are kept at zero level. Also, they provide a good measure of the worst agreement's costs that the firm could accept in order to make up the pool.

4. Implementation issues.

The extended soft constraints model described in previous sections has been implemented in **PETS**. The hardware requirements for this system are very low. Specifically:

- An IBM PC compatible computer running under DOS 3.x.
- 640 Kb of RAM.
- 80x87 floating point unit.
- Hard disk drive (recommended).

The current version of the optimizer requires a minimum of 330 Kb of free RAM. In this environment, maximum problem sizes are 200 rows, 400 variables and 15,000 nonzero entries in the matrix pool. These capabilities have been enough to cover the planning needs of 'El Corte Inglés' over the last two years.

The response of the system is fast enough to allow for a reasonable degree of interactivity. Total execution times for several testcases are shown in Table 4.1:

Case	Time (sec)
1	426
2	404
3	401
4	397
5	371
6	423
7	348
8	405
9	315
10	328
11	524
12	364
13	384
14	333
Avg:	387

Table 4.1.

In all the cases we consider 136 selectable shifts and 21 time intervals. Also, the nonlinear penalty is implemented in three parts. Problem sizes are thus 23 constraints, 63 bounds and

199 integer variables. These times have been measured in an IBM PC XT, with a 4.77 MHz 8088 CPU, a standard floating point coprocessor and a 80 msec. hard drive. Much better performance can be obtained if the system is run in a more powerful platform.

The optimizing algorithm is Gomory's Method of Integer Forms (MIF)[6]. This choice may be surprising because:

- Practical experiences indicate that MIF often yields large convergence times, even for small problems.
- This is a dual method, which does not provide an integer feasible solution up to the last iteration.

However, MIF has proven to be very efficient in our application. As a matter of fact, the program has been runned hundreds or even thousands of times. It has never failed or had running times very different from those indicated in table 4.1.

We have not a theoretical explanation for such a good performance. However, if we take a look at the inverse of the basis in the continuous solution, we will see that it contains only values equal to 0, 1 or -1. In this case, the continuous solution is directly integer because the RHS of the problem is a vector of integer numbers. When a different value appears in the inverse, it is a very "nice" fraction like 1/2, 1/3, 2/3, etc. That may be the reason why Gomory's algorithm behaves so well. It is an open question which deserves some more research.

[6] Vid. Gomory (1963).

5. Conclusions.

In this paper, we have shown that:

- A shift assignment model should distribute the misachievements in workload coverage smoothly among all the time intervals. We have achieved this by means of a nonlinear penalty function.
- The same formulations can be used to multidepartment planning, providing joint efficient solutions.
- Strictly optimal plans for real size models can be computed in a PC compatible environment. This claim contrasts strongly with previous work in this line and, namely, with the **CHEOPS** system[7]. In these approaches, mainframes are required even when optimality is not granted.

PETS is especially suited for applications in department stores, airports, gas stations, hospitals etc. Practical experience shows that the system can produce substantial savings while maintaining (or improving) the quality of service. In Table 5.1 we show a comparison of results between manual and '**PETS**' planning of ten departments from 'El Corte Inglés'.

	MANUAL (A)		PETS (B)		Relative Underach. (B)/(A)	Relative Cost (B)/(A)
Case	**Underach. (hours)**	**Cost (hours)**	**Underach. (hours)**	**Cost (hours)**		
1	17.50	98.75	13.50	98.00	77.14%	99.24%
2	2.00	31.75	0.50	29.00	25.00%	91.34%
3	6.50	86.25	0.00	86.00	0.00%	99.71%
4	4.50	21.25	4.00	20.25	88.89%	95.29%
5	3.00	37.50	3.00	37.50	100.00%	100.00%
6	10.00	105.50	9.00	100.00	90.00%	94.79%
7	8.00	117.00	6.50	110.00	81.25%	94.02%
8	4.50	37.25	4.00	36.25	88.89%	97.32%
9	7.00	21.25	4.00	21.00	57.14%	98.82%
10	0.00	40.00	0.00	40.00	100.00%	100.00%
Avg.	6.30	59.65	4.45	57.80	70.83%	97.05%

Table 5.1: Manual planning vs. PETS.

[7] See Jaquet-Lagrèze and Meziani (1988).

So, for this sample, variable cost savings average around 3% and achievement of quality goals has improved almost 30%. It should be noted that the main target of the firm is service quality. Also, there are other indirect gains: faster planning times, clerical simplification, etc.

According to our experience, the benefits provided by the system depend on several factors:

- **Variability** of personnel requirements over the day. More variability implies higher savings.
- **Availability of workers** from other units with a complementary workload seasonality.
- The **number of available shifts**. A wider set of shifts implies higher savings.
- The **size of the workforce**. A bigger workforce implies higher savings.
- The **number of working hours in a day**. The optimal case is 24 hours service.
- The **length of the planning time period. PETS** can be used to plan workload coverage in fractions of an hour. Shorter periods usually increase the savings.
- The **accuracy of workload forecasts**. This is an essential factor to guarantee a high level of quality in the service.

It should be emphasized that the system will achieve optimal performance when an expert (usually the same person who was in charge of manual planning) controls the software. The conjunction of computing power and the know-how of a seasoned decisionmaker usually yields better achievement of corporate goals. At the same time, using **PETS** the firm becomes less dependent on the abilities of particular individuals.

BIBLIOTECA UCM
5306533403

References.

Gomory, R.E. (1963): 'An Algorithm for Integer Solutions to Linear Programs'. In R.L. Graves and P. Wolfe (eds), '<u>Recent Advances in Mathematical Programming</u>', Mc Graw-Hill, New York. pp. 269-302.

Jaquet-Lagrèze, E.; Meziani, R. (1988): 'Linear Programming and Interactivity: A manpower scheduling DSS'. <u>Cahiers du LAMSADE</u>, nº 84. Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision. Université de Paris-Dauphine. Paris.

Lee, Sang M. (1972): 'Goal Programming for Decision Analysis'. <u>Auerbach Publishers</u>, Philadelphia.